Departamento de Computación
FCEFQyN, Universidad Nacional de Río Cuarto
Asignatura: Estructuras de Datos y Algoritmos - Algoritmos I
Segundo Cuatrimestre de 2016

Práctica No. 7 (Grafos)

Ej. 1. Dado el siguiente grafo dirigido con costos implementado con una matriz de adyacencias:

$$\begin{bmatrix} 1 & 3 & nil & 7 & 8 & nil \\ nil & nil & nil & 18 & nil & nil \\ 7 & nil & 1 & 3 & nil & 15 \\ 1 & 4 & 2 & nil & 7 & 8 \\ nil & 5 & nil & nil & 6 & 9 \end{bmatrix}$$

- Obtenga la representación con listas de adyacencias.
- Aplique Depht-First Search y Breadth-First Search al grafo.
- Aplique el algoritmo de Floyd-Warshall al grafo.
- Aplique el algoritmo de Dijkstra al grafo

Ej. 2. Implementar la clase Graph, con las siguientes operaciones:

- Crear un grafo vacío.
- Decir si un grafo es vaío.
- Dar el número de vertices de un grafo.
- Dar el número de arcos de un frafo.
- Determinar si hay un arco entre dos nodos dados.
- Insertar un nodo en un grafo.
- Insertar un arco entre dos nodos.
- Borrar un vertice del grafo.
- Borrar un arco del grafo.
- Decir si un nodo pertenece al grafo.

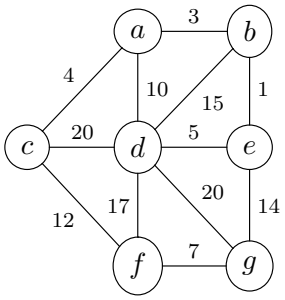
Ej. 3. Implementar el Depht-First Search y el Breadth-First Search en la clase Graph.

Ej. 4. Modifique el algoritmo DFS para hacer un algoritmo que cuente la cantidad de **caminos simples** entre dos nodos dados en un grado dirigido. Cuál es el tiempo de ejecución de su algoritmo.

Ej.5. Haga un algoritmo que, dado un grafo no dirigido, diga si el grafo es conexo. Cuál es el tiempo de ejecución de su algoritmo?

Ej. 6. Dado un grafo dirigido con costos. Proponga un algoritmo que detecte si hay ciclos con costos negativos. Cuál es el tiempo de ejecución del algoritmo?

Ej. 7. Considere el siguiente grafo no dirigido y conexo:



- Utilice el algoritmo de Prims para obtener un árbol abarcador mínimo.
- Utilice el algoritmo de Kruskal para obtener un árbol abarcador mńimo.

Ej. 8. Modificar la clase dada en el ejercicio anterior para incorporar costos en los arcos.

Ej. 9. Implementar el Algoritmo de Warshall en la clase Graph.